



2014

KM-Editor's User Manual

The screenshot shows the KM-Editor interface. On the left is a project tree for 'Project_Smart_INSA' with nodes like 'Main', 'Stage_3', 'Calcul_constitutive', 'Seismic_007', 'Modele_Local', 'Ancien_calcul', 'Draft file', and 'Job'. The main window displays a code editor with a snippet of a file named 'Seismic_009.dgbi'. The code includes various directives and parameters, with some lines highlighted in yellow. A context menu is open over the line 'DEADFONC'. To the right of the code editor is a 'Thematic Help' panel titled 'Title: Sert à préciser les options générales de calcul.' It lists 'OPTION' and 'NOTICE' directives with their descriptions and possible values. The 'OPTION' section includes entries for 'MOT1' (number of assistants), 'MOT2' (color of the frame), 'ASSI' (number of assistants), 'CADR' (positive floating point), 'COSC' (color of the background), 'COUL' (color palette), 'DIM' (dimension of space), 'DONN' (logical unit number or file name), 'ECRA' (echo), 'ELEM' (element to fabricate), 'EPTR' (trace thickness), 'EPSI' (method of calculation), and 'ERRE' (error type). The 'NOTICE' section includes entries for 'KICH' (date and time) and 'DATE' (date).

By : Khuong Le Nguyen

KM-Editor

Version : 2.0

Contents

1	About KM-Editor.....	2
2	How do you manage your code project with KM-Editor	3
2.1	Data synchronization and working on multiple computers	3
2.2	File search.....	3
2.3	Project management	4
2.4	Dynamic help.....	5
3	Modern text editor.....	6
3.1	Table of content	6
3.2	Syntax highlighting and autocomplete	7
3.3	Auto-Indent and Code Folding	8
3.4	Special purpose for Cast3M	8
3.5	Run files continuously	9
3.6	Two windows	10
4	Code Template	11
4.1	Retrieving a stored source code.....	11
4.2	Save your favorite code to the code template library	12
4.3	Synchronize your data library.....	13
5	Perspective and future application with KM-Editor	14

1 About KM-Editor

When I started my thesis at INSA of Lyon in French, my professor asked me to use Cast3M¹ - one of the most population programs for modeling structure. I was very impressed with Cast3M's performance from his calculation results. He showed me the source code that he wrote for a seismic calculating Cast3M on Linux system. I was absolutely shocked from looking at 2500 lines of undocumented code on a terminal screen. After spending sometimes working with Cast3M, I soon realized that Cast3M does not have any good Integrated Development Environment (IDE) or a simple source code editor. Furthermore, after searching on Google I managed to find some related information on Cast3M [wikipedia](#) «Contrairement à d'autres logiciels d'éléments finis, Cast3M ne possède pas d'interface graphique permettant de poser le problème désiré facilement... Cast3M ne possède pas d'éditeur de texte propre mais plusieurs modules ont été développés dans des éditeurs libres pour faciliter son utilisation... ». To sum it all up, Cast3M has neither a Graphical User Interface (GUI) nor its own source code editor. After spending more time digging through the official website of Cast3M, I was happy to discover a few tools to help writing Cast3M source code slightly less challenging such as Crimson Editor or NotePad++. These tools were great help for me at the beginning when I only wrote some small files of code to understand how the program works. After spending more time in Cast3M, my understanding of Cast3M got better and I could write a much longer code for more complicated calculation.

The story of how KM-Editor became what it is today was all thanks to my parametric studies and some projects that I got involved in during the first period of my thesis. With the parametric study, I had to write multiple code files and execute one after another. I knew some techniques to run multiple code files continuously on Linux and Window but they are not quite efficient. In addition, if you work with Cast3M in a big project, you will agree with me that there are only a few files used frequently but they are not easy to find in a project directory which includes hundreds of files (inputs files, calculations files, output files...). So my question was how I could manage my project well and make the calculation easier. And that was the reason why I took my time at home to create KM-Editor.

In this guide, I will show you what KM-Editor V2.0.1 is capable of and how it will help you to work more efficiently with Cast3M.

¹Cast3M: is open source software, free for the education and researching etude. The main guide of Cast3M could be found in the office website of Cast3M or my website [KM-Editor.com](#).

2 How do you manage your code project with KM-Editor

2.1 Data synchronization and working on multiple computers



For working on a big project, you usually need to collaborate with your colleagues and divide the work into several parts. The working project should be updated automatically and the data should be accessible from everywhere at any time. To meet this requirement, I would suggest you use one of these cloud services for data synchronization: Google Drive, Dropbox, SkyDriver. Having your project files sync to these cloud services mean that your updates will be transferred between yours computers automatically. After that, you just need to install KM-Editor on each working computer to open the project file. KM-Editor will automatically recognize all file within the project.

If you want to use an USB instead of these programs, the principle is the same.

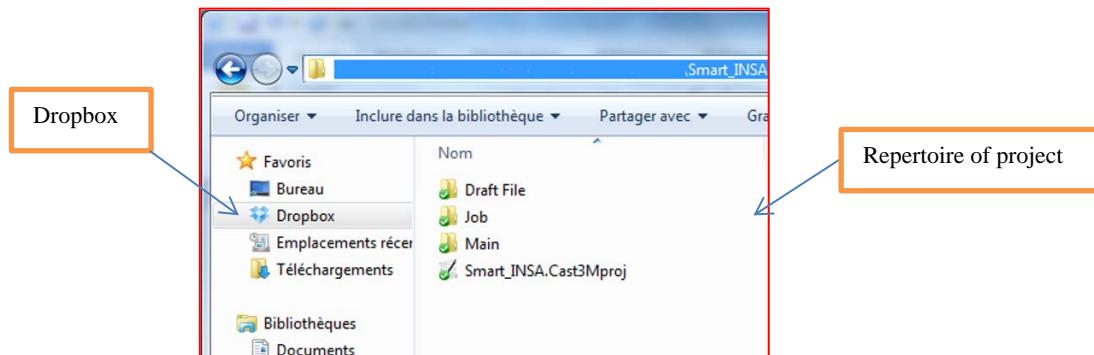


Figure 1: Synchronize data with Dropbox

2.2 File search

Sometimes, you may want to search for a particular file which you know the name partially. You could achieve this by using the “Files Search” tool. You will need to pay close attention to the search directory. By default, KM-Editor selects the directory of the selected file in the tree view of project.

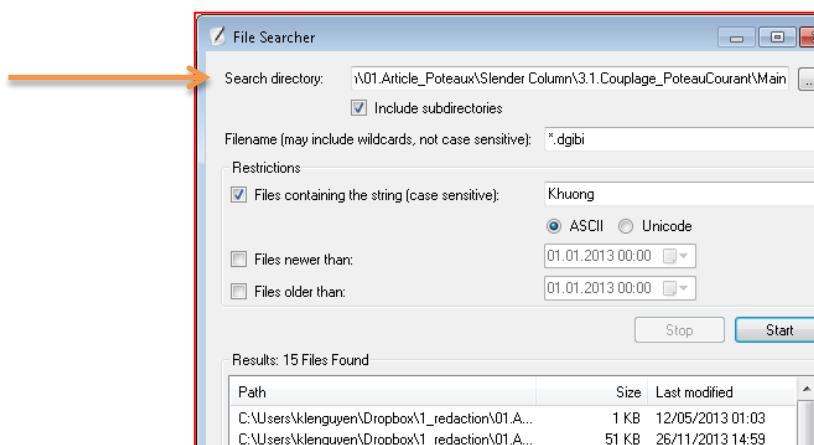
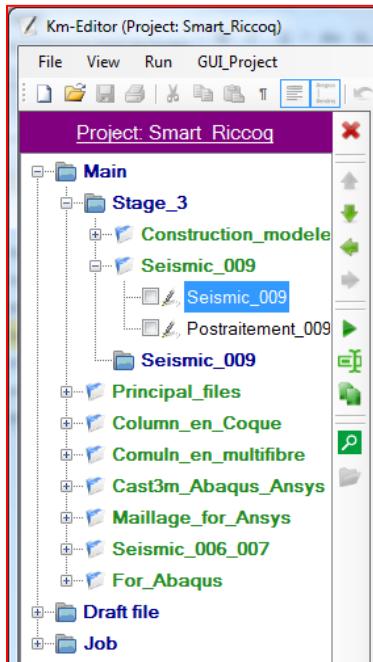


Figure 2: File Search Utility

2.3 Project management



As I mentioned above, KM-Editor allows you to manage your project smartly. You don't have to take your time to memorize or look for an important file in a folder which includes hundreds of files.

For each project with KM-Editor, you have three main directories:

- Main: to manage the principal files that you want to work frequently.
- Draft file: to manage the draft files.
- Job: to stock the job's repertoires.

For these three main directories, I fixed their name by default so that you can't rename but you can do everything with their sub-repertoires.

Figure 3: Project structure

For more information, I created a video to show you how to create and manage a project:

<https://www.youtube.com/watch?v=DQXBahlKpGk>

In the video, I used the “Add” command to build my project. They are very easy to use and just take a few minutes to get familiar with.

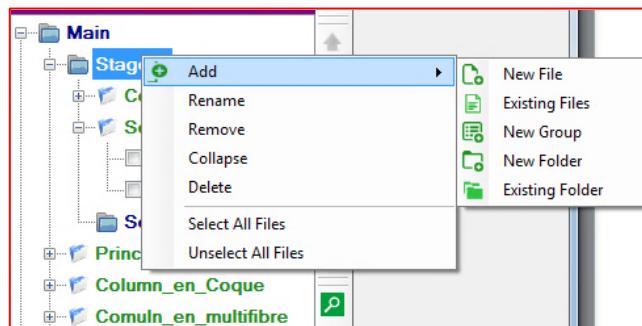


Figure 4: Add tools

Take some minutes to arrange your project now and you will save a lot of time tomorrow.

2.4 Dynamic help

KM-Editor for Cast3M includes most of the thematic notes of Cast3M. You can easily search the help for any keyword by a simple operation: select the keyword (by double click your left mouse for example) → right click and select “Help + “keyword”” to see the help.

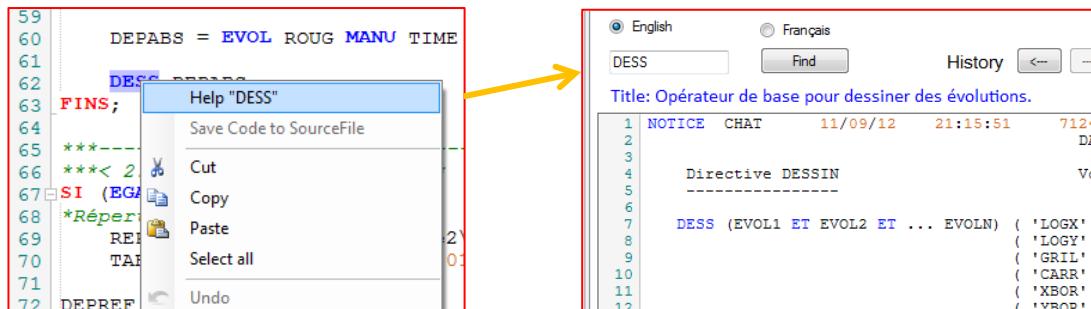


Figure 5: Consultation of keyword notice

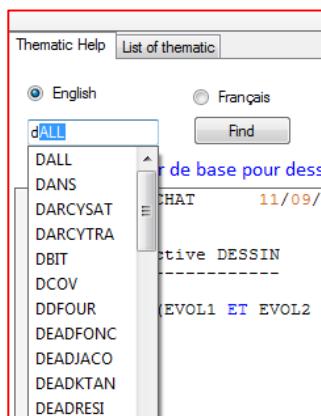


Figure 6: Keyword search

When you search a keyword in “Thematic Help” window box, you are also offered with autocompletion as shown in the screenshot. It's useful when you don't memorize exactly how a keyword is written.

You can also find a keyword in the “List of thematic” window where they had been organized by thematic.

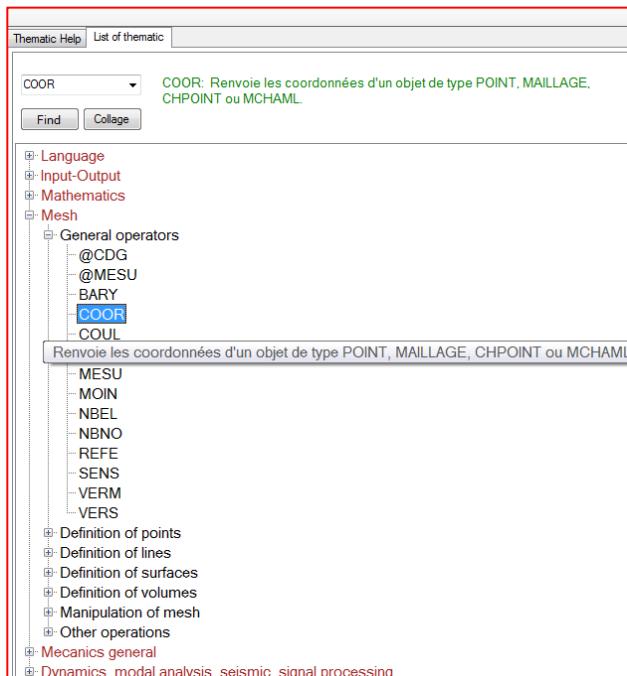


Figure 7: Thematic of keyword

3 Modern text editor

KM-Editor is a modern text editor including a lot of features to help you write your code. I will not talk about his basic features like “Copy”, “Paste”, “Undo”, “Redo”, “Find”, “Replace”... In this guide, I will try to demonstrate the main features that I had developed for Cast3M.

3.1 Table of content

Although the project management tool helps you a lot in organizing the project source codes, there will be times when you could not remember where a particular piece of code is. With KM-Editor, You could easily create an index to look up later by using the “Structure of File” function.

To understand how this works, you can choose any code file and put a special comment code block to define a indexed object. For example, the code block `***< Section: section A >***` indicates that you are starting “section A”. In KM-Editor, You could define four types of indexed objects such as “Section”, “Model”, “Mesh” and “Others”. Note that these code block are not executed during runtime of the application as it follows Cast3M commented code block standard.

In this video “[Table of contents](#)”, you can see how to define an index table for a long code file of 1773 lines. Once you have the index in place, it is much easier to find and change the source code in the future.

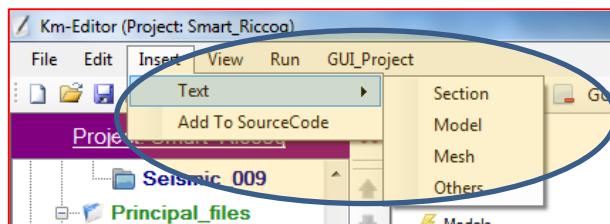


Figure 8: Adding remarks utility

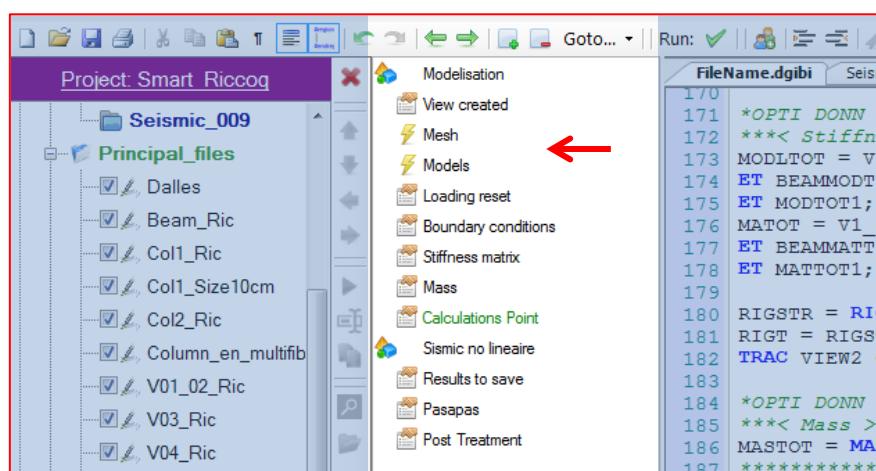
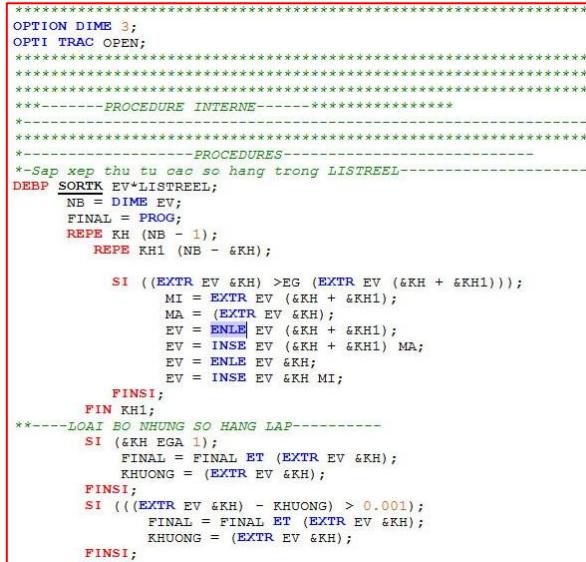


Figure 9: Table content of text code.

3.2 Syntax highlighting and autocomplete

KM-Editor comes with syntax highlighting. This function helps you quickly identify any mistakes you may have made while typing the code and the code is much more readable.



```

***** OPTION DIME 3;
***** OPTI TRAC OPEN;
***** -----
***** PROCEDURE INTERNE -----
***** -----
***** PROCEDURES
*-- Sap xep thu tu cac so hang trong LISTREEL
DEBP SORTK EV+LISTREEL;
NB = DIME EV;
FINAL = PROG;
REPE KH (NB - 1);
REPE KH1 (NB - &KH);

SI ((EXTR EV &KH) >= EG (EXTR EV (&KH + &KH1)));
    MI = EXTR EV (&KH + &KH1);
    MA = (EXTR EV &KH);
    EV = ENLE EV (&KH + &KH1);
    EV = INSE EV (&KH + &KH1) MA;
    EV = ENLE EV &KH;
    EV = INSE EV &KH MI;

FINSI;
PIN KH1;
-----DOAT BO NHUNG SO HANG LAP-----
SI (&KH EGA 1);
    FINAL = FINAL ET (EXTR EV &KH);
    KHUONG = (EXTR EV &KH);
FINSI;
SI (((EXTR EV &KH) - KHUONG) > 0.001);
    FINAL = FINAL ET (EXTR EV &KH);
    KHUONG = (EXTR EV &KH);
FINSI;

```

Figure 10: Syntax highlight

Another great feature of KM-Editor is the ability to autocomplete while you're typing the code. This feature will help you code in Cast3M much faster and more efficient. Figure 11 demonstrates how KM-Editor gives you all the possible functions that could be used starting with DE. At the moment, KM-Editor's suggestions are in French. In the future, I will try to make it available in English.

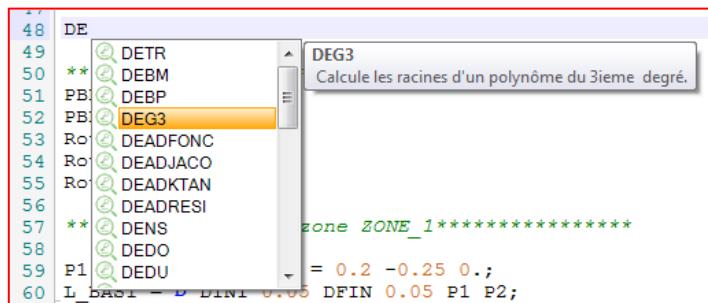


Figure 11: Auto-complete keyword

3.3 Auto-Indent and Code Folding

The “Auto-Indent” and “Code folding” tools help you write your code more clearly and more readable.

```

23 DEBP ACIMULT MAI1*MAILLAGE DIR1*ENTIER NB*ENTIER
24 *Direction OX
25 SI (EGA DIR1 1);
26 REPE KH1 NB;
27 SI (EGA &KH1 1);
28 MAI1T = MAI1;
29 SINO;
30 MAI1T = MAI1T ET
31 (MAI1 PLUS ((&KH1 - 1)*ESPA1) 0. 0.));
32 FINS;
33 FIN KH1;
34 FINS;
35 *Direction OY
36 SI (EGA DIR1 2);
37 REPE KH1 NB;
38 SI (EGA &KH1 1);
39 MAI1T = MAI1;
40 SINO;
41 MAI1T = MAI1T ET
42 (MAI1 PLUS (0. ((&KH1 - 1.)*ESPA1) 0.));
43 FINS;
44 FIN KH1;
45 FINS;
46 FINP MAI1T;

```

```

23 DEBP ACIMULT MAI1*MAILLAGE DIR1*ENTIER NB*ENTIER ESPA1*FI
24 *Direction OX
25 SI (EGA DIR1 1);
26 REPE KH1 NB;
27 SI (EGA &KH1 1);
28 MAI1T = MAI1;
29 SINO;
30 MAI1T = MAI1T ET
31 (MAI1 PLUS ((&KH1 - 1)*ESPA1) 0. 0.));
32 FINS;
33 FIN KH1;
34 FINS;
35 *Direction OY
36 SI (EGA DIR1 2);
37 REPE KH1 NB;
38 SI (EGA &KH1 1);
39 MAI1T = MAI1;
40 SINO;
41 MAI1T = MAI1T ET
42 (MAI1 PLUS (0. ((&KH1 - 1.)*ESPA1) 0.));
43 FINS;
44 FIN KH1;
45 FINS;
46 FINP MAI1T;

```

Figure 12: Auto-Indent

```

14 ****
15 DEBP METTR3D MAI*MAILLAGE
16 PO P1*POINT
17 TORX TORY TORZ*FLOTTANT;
18
19 X0 Y0 Z0 = COOR P0;
20 X1 Y1 Z1 = COOR P1;
21
22 MAI1 = MAI1 PLUS ((X1 - X0) (Y1 - Y0) (Z1 - Z0));
23
24 MAI1 = MAI1 TOUR TORX P1 (P1 PLUS (5. 0. 0.));
25 MAI1 = MAI1 TOUR TORY P1 (P1 PLUS (0. 5. 0.));
26 MAI1 = MAI1 TOUR TORZ P1 (P1 PLUS (0. 0. 5.));
27
28 FINP MAI1;
29
30 DEBP DALLK PK*POINT LX1 LY1 LY2 MAILX MAILY*FLOTTANT;
51

```

```

14 ****
15 DEBP METTR3D MAI*MAILLAGE
29
30 DEBP DALLK PK*POINT LX1 LY1 LY2 MAILX MAILY*FLOTTANT;
51

```

Figure 13: Code folding

3.4 Special purpose for Cast3M

As you know, Cast3M only compile the first 72 characters of a line. KM-Editor for Cast3M was designed to help user distinguish when the first 72 characters ends. The executing codes have a purple background while the rest of the commented codes are green color on white background. The code illustrated in Figure 14 is an example.

```

CHAM1 = MANU CHML MODTOT BTRA 1. TYPE CARACTERISTIQUES RIGIDITE; All letter in this place considered like comment
CHAM2 = INTG CHAM1 MODTOT ELEM BTRA;
TAIL1 = (2 ** (1.0 / 2.0)) * CHAM2 ** (1.0 / 2.0);

```

Figure 14: Distinction two places of line code

3.5 Run files continuously

KM-Editor could run a series of codes continuously. For this to work, you could remove the line “PAUSE” at the end of file “castem_version.bat”. The “PAUSE” line is useful when you want to see the messages displayed at the end of a running Cast3M process like Figure 15 but it does not allow you to run several files continuously.

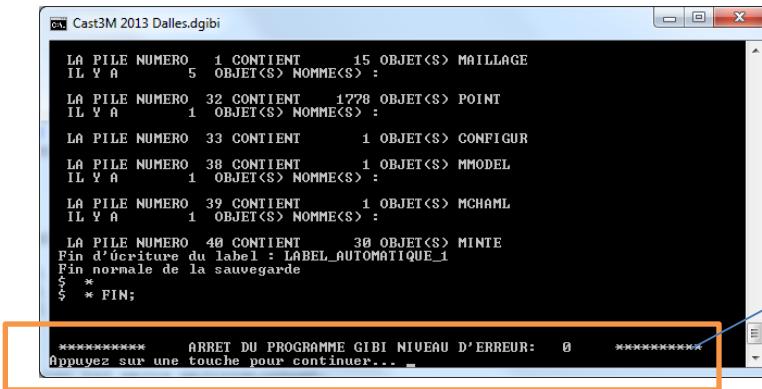


Figure 15: Running Cast3M Window

You have two methods to run a series of code file that are demonstrated in Figure 16. The first method executes the selected files continuously and save the results in the same directory. The second method executes the selected jobs continuously and save the results in the job's directory.

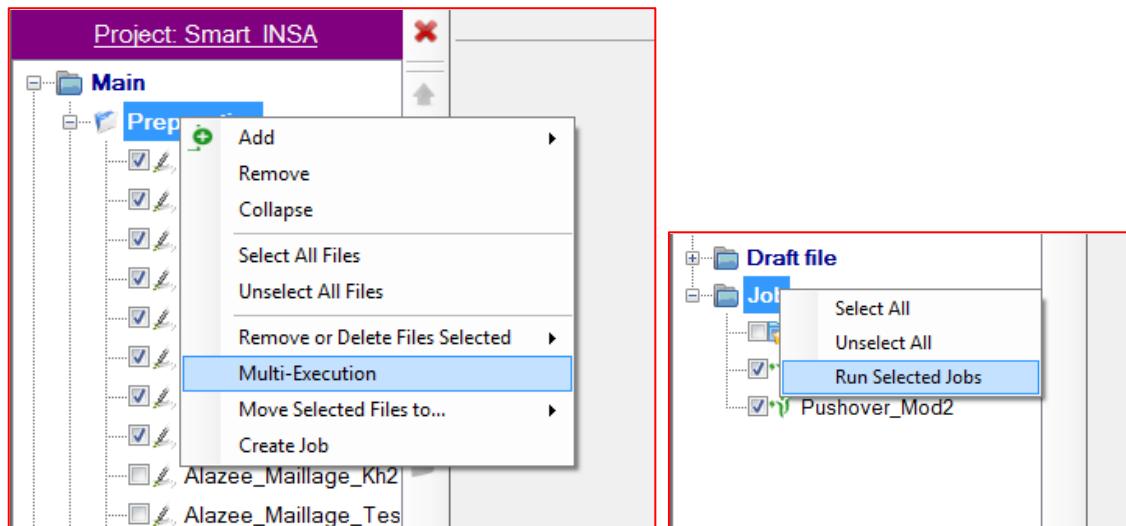
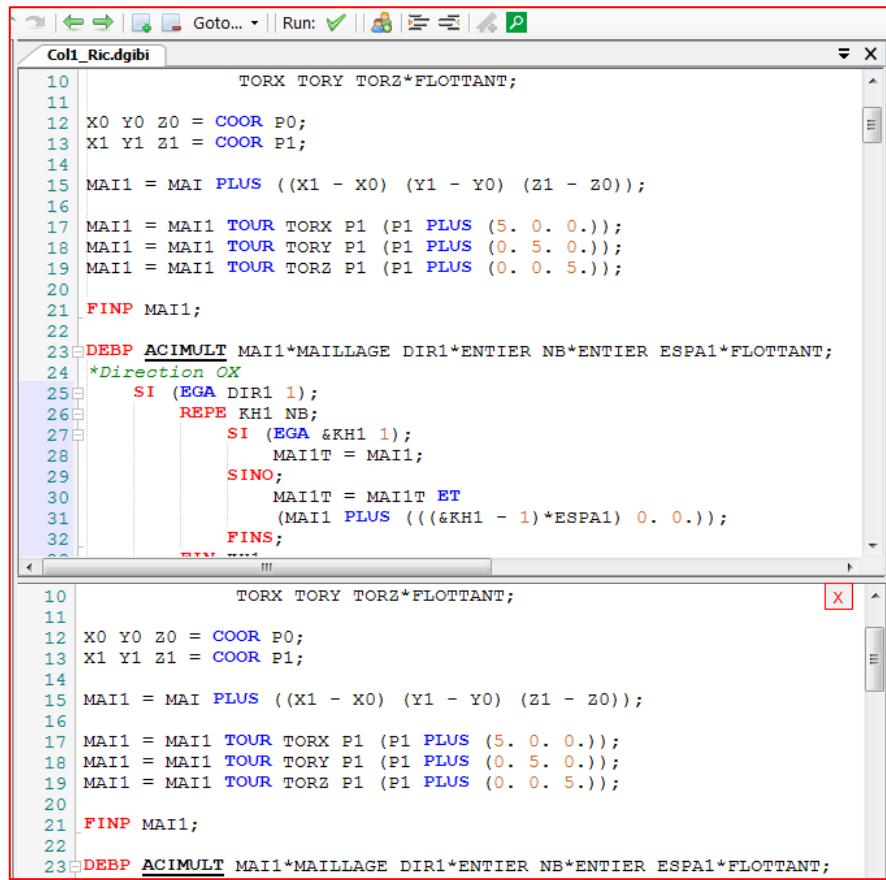


Figure 16: Two methods to run several file continuously

3.6 Two windows

Like Word Office, KM-Editor allows you to see two different places of the same file at the same time.



```

10          TORX TORY TORZ*FLOTTANT;
11
12 X0 Y0 Z0 = COOR P0;
13 X1 Y1 Z1 = COOR P1;
14
15 MAI1 = MAI1 PLUS ((X1 - X0) (Y1 - Y0) (Z1 - Z0));
16
17 MAI1 = MAI1 TOUR TORX P1 (P1 PLUS (5. 0. 0.));
18 MAI1 = MAI1 TOUR TORY P1 (P1 PLUS (0. 5. 0.));
19 MAI1 = MAI1 TOUR TORZ P1 (P1 PLUS (0. 0. 5.));
20
21 FINP MAI1;
22
23 DEBP ACIMULT MAI1*MAILLAGE DIR1*ENTIER NB*ENTIER ESPA1*FLOTTANT;
24 *Direction OX
25 SI (EGA DIR1 1);
26 REPE KH1 NB;
27 SI (EGA &KH1 1);
28 MAI1T = MAI1;
29 SINO;
30 MAI1T = MAI1T ET
31 (MAI1 PLUS (((&KH1 - 1)*ESPA1) 0. 0.));
32 FINS;

```

```

10          TORX TORY TORZ*FLOTTANT;
11
12 X0 Y0 Z0 = COOR P0;
13 X1 Y1 Z1 = COOR P1;
14
15 MAI1 = MAI1 PLUS ((X1 - X0) (Y1 - Y0) (Z1 - Z0));
16
17 MAI1 = MAI1 TOUR TORX P1 (P1 PLUS (5. 0. 0.));
18 MAI1 = MAI1 TOUR TORY P1 (P1 PLUS (0. 5. 0.));
19 MAI1 = MAI1 TOUR TORZ P1 (P1 PLUS (0. 0. 5.));
20
21 FINP MAI1;
22
23 DEBP ACIMULT MAI1*MAILLAGE DIR1*ENTIER NB*ENTIER ESPA1*FLOTTANT;

```

Figure 17: 2 views for a code text

4 Code Template

The Code Template tool in KM-Editor is a great tool to store your favorite piece of code for Cast3M and reuse in other projects. In this section, I will show you how to create a code to save in library and how to access an existing code within the code template library.

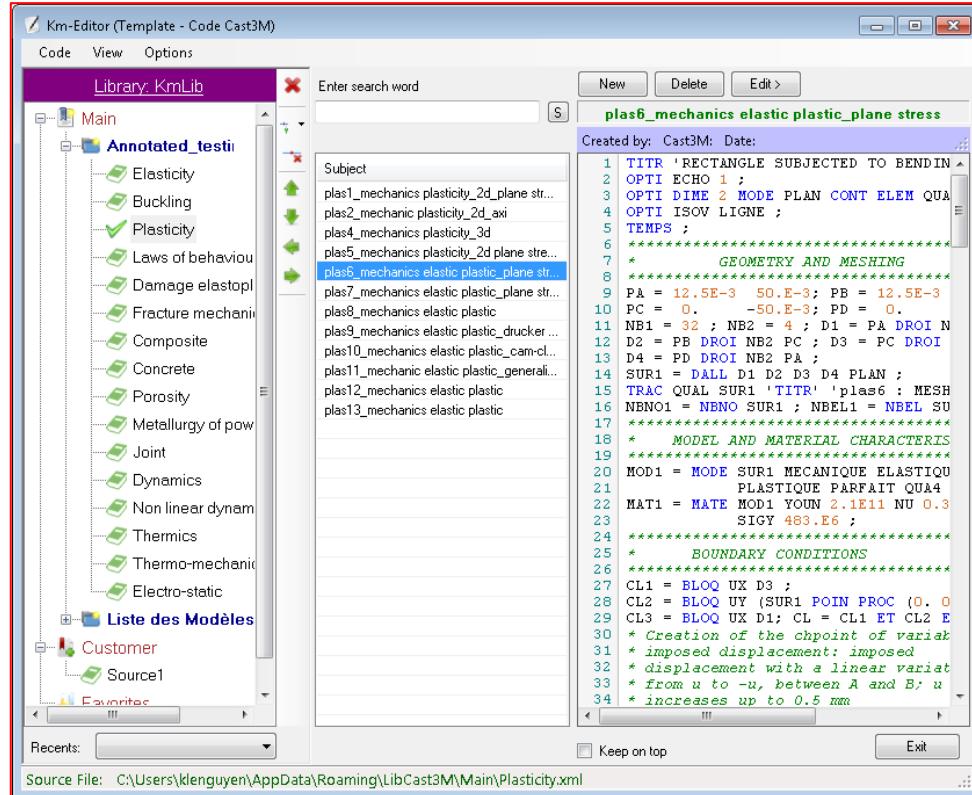


Figure 18: Main windows of code library

4.1 Retrieving a stored source code

In Figure 18, the code template library tool has three working tabs. The left tab expose you all source files of library. On selecting a source file, the content will be displayed in the middle tab. And the right tab shows information of selected item. You can access the library directly from the main window of KM-Editor or from the project's window.

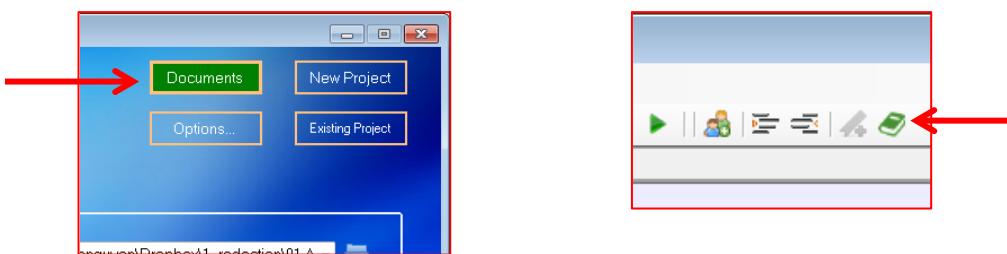


Figure 19: Two ways to open the code library window

4.2 Save your favorite code to the code template library

When you are writing some codes, you may want to save it to the code template library and use in the future, it's as simple as select this code paragraph, right click and select "Save code to Source File". This will immediately prompt the user to save the selected code into the library. Note that the "Source Activating" path is where the template of the source code will be stored. You can change this path by selecting another source file in the treeview of project library shown in the left tab.

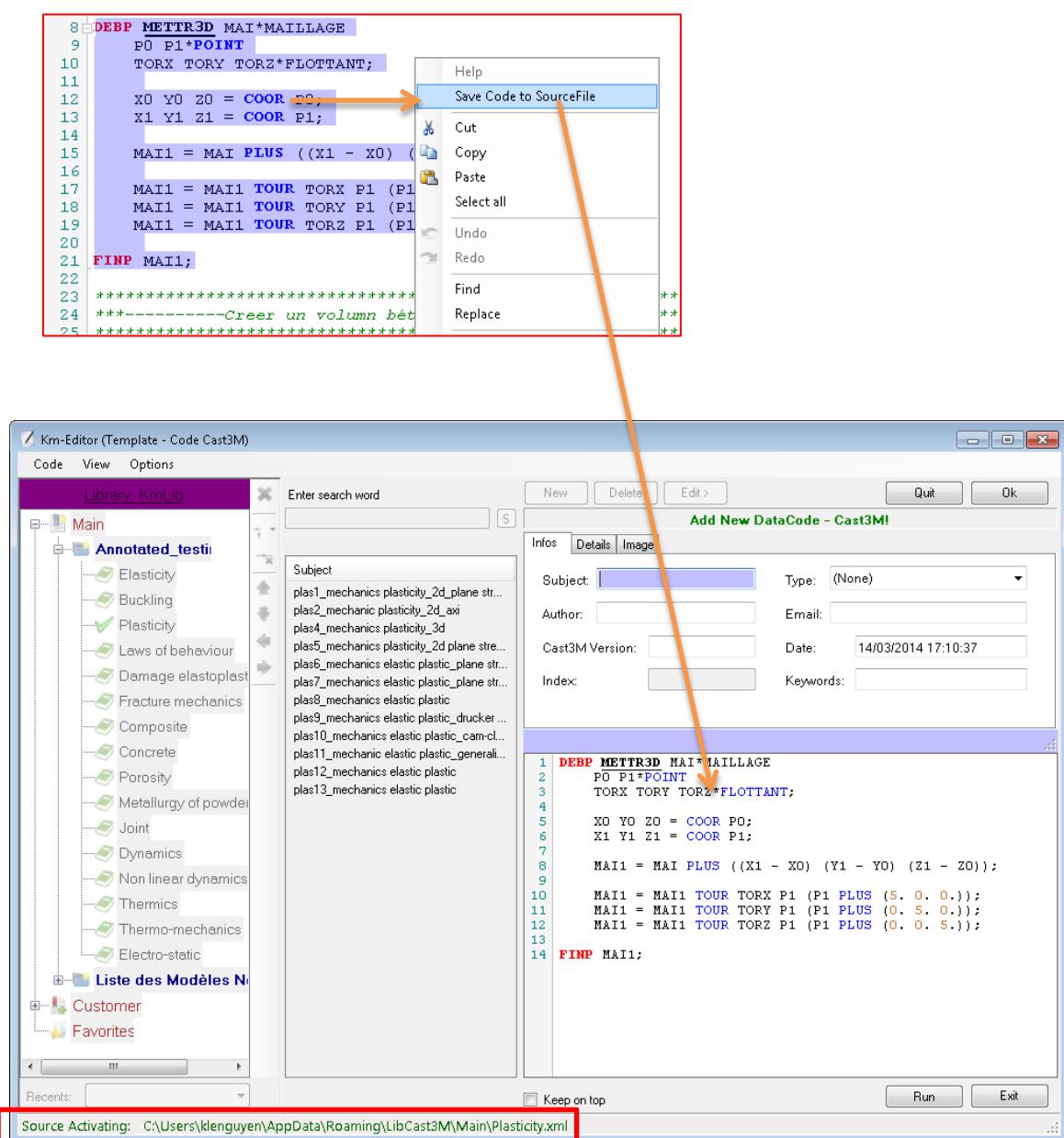


Figure 20: Save your code to library

4.3 Synchronize your data library

As I had shown you in the section 2.1 of this guide, we could use a program like Dropbox or Google Drive to synchronize the project and also our data library. By default, the data library is installed in the program's directory (..|AppData|Roaming|), you can copy this directory to Dropbox, redefine the path of .Cast3Mlib file like Figure 21 and then your library will be synchronized between your computers.

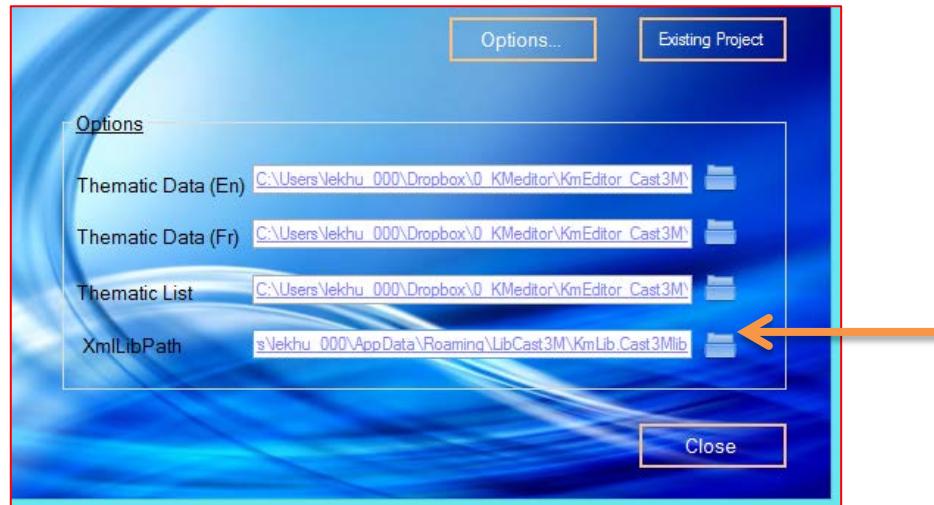


Figure 21: Define the library path

You can also see the demonstration in this video:

<https://www.youtube.com/watch?v=b4i7mPbfVME>

5 Perspective and future application with KM-Editor

After reading this User's Manual of KM-Editor, I hope you will find KM-Editor useful and use it as your favorite tool to practice and manage your Cast3M project. At the moment, I want to offer the program to all student and researcher who use Cast3M as a tool of scientific computing.

In the next version, I am hoping to develop KM-Editor for others languages such as Matlab, Python, Abaqus.Inp... and make it available on Mac OS X and Linux systems.

Thank you for your attention!

To join us online:

- <http://www.km-editor.com/>
- <https://www.facebook.com/pages/KM-Editor/504535432922530>

